Module 7: Matching and Weighting Estimators

Fall 2021

Matthew Blackwell

Gov 2003 (Harvard)

1/ Matching estimators

The problem with regression

- Causal inference is all about comparing **counterfactuals**, like the ATT:

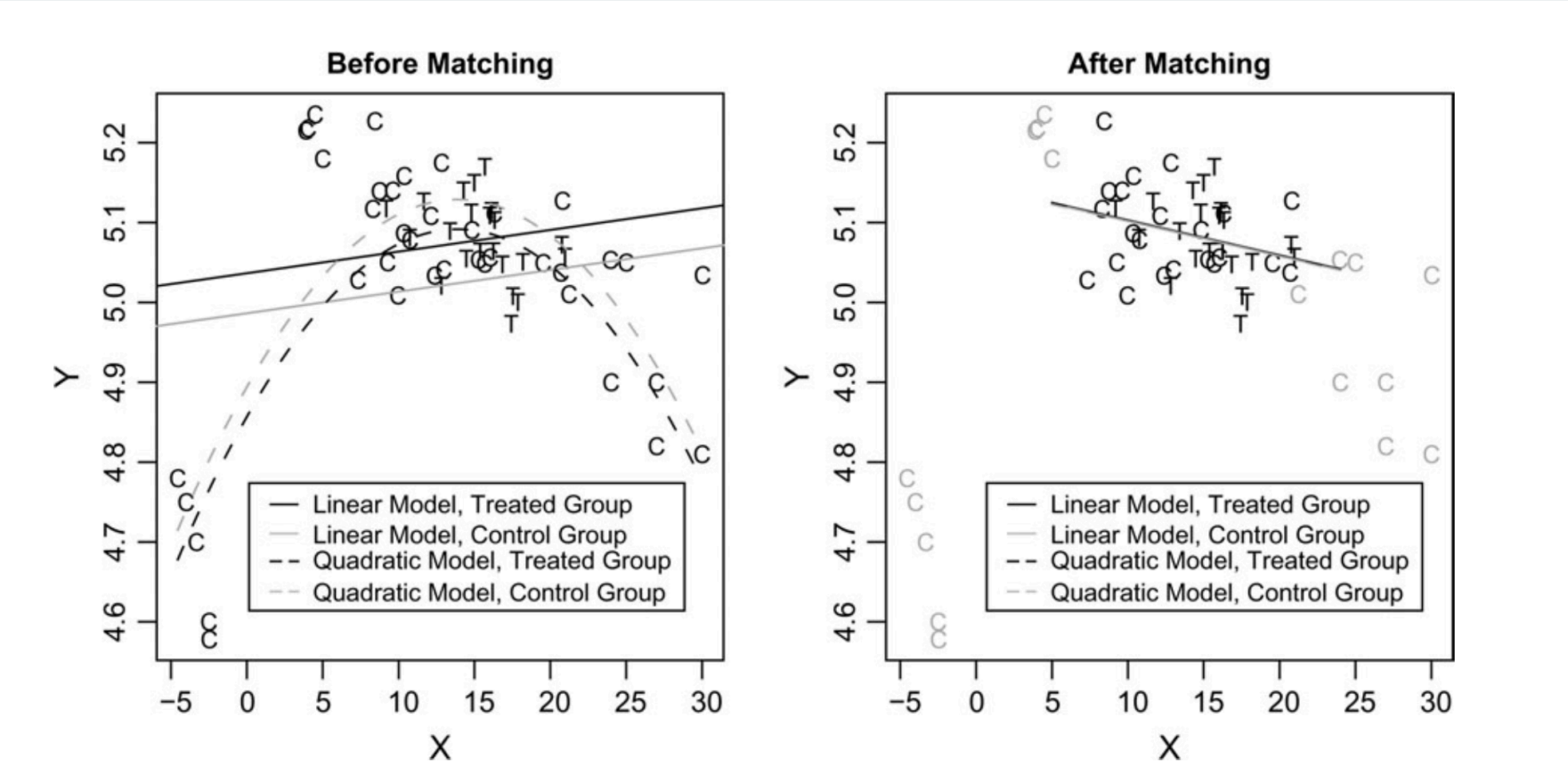
$$\tau_{ATT} = \mathbb{E}[Y_i(1) - Y_i(0) \mid D_i = 1]$$

- Recall the **imputation** estimators with regression.

$$\widehat{\tau}_{\text{reg}} = \frac{1}{n_1} \sum_{i=1}^{n} D_i (Y_i - \hat{\mu}_0(\mathbf{X}_i))$$

- Common solution: use a parametric model for $\hat{\mu}_0(\mathbf{X}_i)$
  - For example, could assume it is linear: $\mu_0(\mathbf{x}) = \mathbf{x}'\beta$
  - Regression, MLE, Bayes, etc.
  - But this model might be wrong $\rightsquigarrow$ wrong causal estimates.

Model dependence



What is matching?

- **Matching** is a nonparametric imputation estimator:

$$\widehat{\tau}_m = \frac{1}{n_1} \sum_{i=1}^{n} D_i \left(Y_i - \frac{1}{M} \sum_{j \in \mathcal{J}_M(i)} Y_j \right)$$

  - $\mathcal{J}(i)$ are the set of M closest control units to i in terms of $\mathbf{X}_i$

- Matching has strong advantages:

  1. Reduces dependence of estimates on parametric models.
  2. Reduces model-based extrapolation.
  3. Makes counterfactual comparisons more transparent.

- What matching isn't: a solution for selection on unobservables.

  - Matching is an **estimation** technique, not an identification strategy.

Types of matching

- Assumptions:
  - No unmeasured confounders: $D_i \perp\!\!\!\perp (Y_i(0), Y_i(1)) \mid \mathbf{X}_i$
  - Overlap/positivity: $0 < \mathbb{P}(D_i = 1 \mid \mathbf{X}_i = x) < 1$
- **Exact matching**: choose matches that have the same value of $\mathbf{X}_i$
  - $\mathcal{J}_M(i)$ is a random set of M control units with $\mathbf{X}_j = \mathbf{X}_i$
  - Covariate distribution is treated and matched controls exactly the same:

$$\widehat{\mathbb{P}}(\mathbf{X}_i = \mathbf{x} \mid D_i = 1) = \widehat{\mathbb{P}}(\mathbf{X}_j = \mathbf{x} \mid D_j = 0, j \text{ is matched})$$
$$\rightsquigarrow \mathbb{E}[Y_i(0) \mid D_i = 1] = \mathbb{E}[Y_j \mid D_j = 0, j \text{ is matched}]$$

  - Problem: not feasible with high-dimesional or continuous $\mathbf{X}_i$
- **Coarsened exact matching** (Iacus et al, 2011)
  - Discretize and group covariates into substantively meaningful bins
  - Exact match on these bins $\rightsquigarrow$ accounts for interactions
  - Have to drop treated units in bins with no controls $\rightsquigarrow$ changes estimand.
  - Allows you to control bias/variance tradeoff through coarsening.

Matching in high dimensions

- Even CEM can break down with high dimensional $\mathbf{X}_i$.
- We can define closeness using lower dimensional **distance metrics**
  - Reduces dimensionality: maps two vectors to a single number
- **Mahalanobis distance**:

$$D(\mathbf{X}_j, \mathbf{X}_j) = \sqrt{(\mathbf{X}_i - \mathbf{X}_j)' \widehat{\Sigma}^{-1} (\mathbf{X}_i - \mathbf{X}_j)}$$

  - $\widehat{\Sigma}$ is the estimated variance-covariance matrix of the observations:

$$\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T$$

- Estimated propensity score:

$$D(\mathbf{X}_i, \mathbf{X}_j) = |\widehat{\pi}(\mathbf{X}_i) - \widehat{\pi}(\mathbf{X}_j)| = \left|\hat{\mathbb{P}}(D_i = 1 \mid \mathbf{X}_i) - \hat{\mathbb{P}}(D_i = 1 \mid \mathbf{X}_j)\right|$$

  - Some use the linear predictor: $D(\mathbf{X}_i, \mathbf{X}_j) = |\text{logit}(\widehat{\pi}(\mathbf{X}_i)) - \text{logit}(\widehat{\pi}(\mathbf{X}_j))|$

Other matching choices

- **Matching ratio** how many control units per treated?
  - Lower reduces bias (only use the closest matches)
  - Lower increases variance
- **With or without replacement** same control matched to multiple treated?
  - With replacement gives better matches & matching order doesn't matter.
  - Without replacement simplifies variance estimation.
- **Caliper**: drop poor matches?
  - Only keep matches below a distance threshold, $D(\mathbf{X}_i, \mathbf{X}_j) \leq c$
  - Reduces imbalance, but if you drop treated units, estimand changes.

Propensity scores, redux

- Covariates are balanced conditional on true propensity score:

$$D_i \perp\!\!\!\perp \mathbf{X}_i \mid \pi(\mathbf{X}_i)$$

- Implies we only need to match/balance on $\pi(\mathbf{x})$:

$$(Y_i(0), Y_i(1)) \perp\!\!\!\perp D_i \mid \mathbf{X}_i \iff (Y_i(0), Y_i(1)) \perp\!\!\!\perp D_i \mid \pi(\mathbf{X}_i)$$

- In observational data we never know the true $\pi(\mathbf{x}) \rightsquigarrow$ estimate it $\widehat{\pi}(\mathbf{x})$.
- Is balancing on $\widehat{\pi}(\mathbf{x})$ sufficient? **No idea!!**
  - Have to check if $\mathbf{X}_i$ is actually balanced.
  - Somewhat deflates the benefits of PS matching/balancing.
- $\rightsquigarrow$ "propensity score tautology"

Assessing balance

- Goal of matching is to maximize balance: $\widehat{F}_1(\mathbf{x}) \approx \widehat{F}_{0,\mathcal{J}}(\mathbf{x})$
  - Joint distribution of $\mathbf{X}_i$ is similar between treated and matched controls.
  - Difficult to asses balance across many dimensions $\rightsquigarrow$ summaries.
- Options:
  - Differences-in-means/medians, standardized.
  - QQ plots/KS statistics for comparing the entire distribution of X_i.
  - L_1: multivariate histogram (for CEM)
  - Choice of metric can change what matching method works best.
- Hypothesis tests for balance are problematic:
  - Dropping units can lower power ($\uparrow$ p-values) without a change in balance.

Bias of inexact matching

- To show the bias on matching, focus on finding a single control match.
- Let $j(i)$ be the matched control for unit i, the bias is:

$$\mathbb{E}\left[Y_j \mid D_i = 1, \mathbf{X}_i, \mathbf{X}_j\right] - \mathbb{E}[Y_i(0) \mid D_i = 1, \mathbf{X}_i] = \underbrace{(\mu_0(\mathbf{X}_i) - \mu_0(\mathbf{X}_{j(i)}))}_{\text{unit-level bias}}$$

  - Bias is 0 if matching is exact since $\mathbf{X}_i = \mathbf{X}_{j(i)}$
  - Bias grows with **matching discrepancy**/imbalance.

- **Bias correction**: estimate $\hat{\mu}_0(\mathbf{x})$ with regression and estimate bias.

$$\widehat{Y}_i(0) = Y_{j(i)} - (\hat{\mu}_0(\mathbf{X}_i) - \hat{\mu}_0(\mathbf{X}_{j(i)}))$$

  - Imputation of missing potential outcome now matching + regression.
  - Generalizes easily to any number of matches.

Variance

- Matching with replacement: cluster on the match.
  - Can either use clustered SEs or cluster bootstrap.
  - Valid for post-matching regression (Abadie and Spiess, 2021)
- Matching without replacement: more complicated.
  - Same control unit matched to multiple treated: no easy clustering.
  - $K_M(i)$ is the number of times a unit is used as a match
- Assuming units are well-matched so bias can be ignored,

$$\mathbb{V}(\widehat{\tau}_m) = \frac{1}{n_1}\left(\underbrace{\mathbb{E}\left[(\tau(\mathbf{X}_i) - \tau_{ATT})^2 \mid D_i = 1\right]}_{\text{variance of CATE on treated}} + \underbrace{\mathbb{V}[\widehat{\tau}_m \mid \mathbb{X}, \mathbf{D}]}_{\text{conditional variance}}\right)$$

  - Abadie and Imbens (2006) provides matching-based variance estimators.

2/ Weighting estimators

Why weighting?

- Matching has a couple of downsides:
  - Inefficient: it may throw away data.
  - Ineffective: crude tool so it may not be able to balance covariates.
- Matching is actually a special case of a weighting estimator:

$$\begin{aligned}\widehat{\tau}_m &= \frac{1}{n_1}\sum_{i=1}^{n} D_i\left(Y_i - \frac{1}{M}\sum_{j\in\mathcal{J}_M(i)} Y_j\right) \\ &= \frac{1}{n_1}\sum_{i:D_i=1} Y_i - \frac{1}{n_0}\sum_{i:D_i=0} \underbrace{\left(\frac{n_0}{n_1}\frac{K_M(i)}{M}\right)}_{\text{weight}} Y_i\end{aligned}$$

  - $K_M(i)$ is the number of times i is used as a match.
- Weighting estimators choose the weights directly to reduce imbalance.

Survey sampling

- Imagine we want to estimate the population mean $\overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i$
  - $R_i = 1$ if sampled, $R_i = 0$ if not.
  - We only observe Y_i for those with $R_i = 1$.
  - Inclusion probability varies by person: $\mathbb{P}(R_i = 1) = \pi_i$
- **Horvitz-Thompson estimator** is unbiased (treating Y_i as fixed):

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{R_i Y_i}{\pi_i}\right] = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{E}[R_i]Y_i}{\pi_i} = \frac{1}{n}\sum_{i=1}^{n}\frac{\pi_i Y_i}{\pi_i} = \overline{Y}$$

- Key idea: reweight sample to be representative of population.

Horvitz-Thompson for treatment effects

- Applying HT potential outcomes: weight by inverse propensity score.

$$\widehat{\text{ATE}} = \widehat{\tau}_{ipw} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{D_i Y_i}{\widehat{\pi}(\mathbf{X}_i)} - \frac{(1-D_i)Y_i}{1-\widehat{\pi}(\mathbf{X}_i)}\right)$$

- Under no unmeasured confounders, $\mathbb{E}[\widehat{\tau}_{ipw}] \xrightarrow{p} \tau$ (consistent)
  - Would be unbiased if we knew the true propensity scores, $\pi(\mathbf{X}_i)$
- Similar expression for ATT:

$$\widehat{\text{ATT}} = \widehat{\tau}_{ipw,t} = \frac{1}{n_1}\sum_{i=1}^{n}\left(D_i Y_i - \frac{\widehat{\pi}(\mathbf{X}_i)(1-D_i)Y_i}{1-\widehat{\pi}(\mathbf{X}_i)}\right)$$

- Logic: upweight units with "rare" treatment values for their values of $\mathbf{X}_i$
  - A kind of "continuous" version of matching with replacement.

Normalized weights

- HT estimators with known weights are unbiased but can be inefficient.
  - Large weights can lead to highly variable estimates when (not) included.
- **Hajek estimator** normalizes the denominator so the weights sum to 1:

$$\hat{\mu}_h = \hat{\mathbb{E}}[Y_i] = \frac{\sum_{i=1}^n (R_i/\pi_i) Y_i}{\sum_{i=1}^n R_i/\pi_i}$$

- Hajek estimator for the ATE:

$$\widehat{\tau}_h = \frac{\sum_{i=1}^n D_i Y_i/\widehat{\pi}(\mathbf{X}_i)}{\sum_{i=1}^n D_i/\widehat{\pi}(\mathbf{X}_i)} - \frac{\sum_{i=1}^n (1-D_i) Y_i/(1-\widehat{\pi}(\mathbf{X}_i))}{\sum_{i=1}^n (1-D_i)/(1-\widehat{\pi}(\mathbf{X}_i))}$$

- Practically, weighted least squares gives automatic normalization:

$$(\hat{\alpha}_{\text{wls}}, \widehat{\tau}_{\text{wls}}) = \arg\min_{\alpha,\tau} \sum_{i=1}^n \left(\frac{D_i}{\widehat{\pi}(\mathbf{X}_i)} + \frac{1-D_i}{1-\widehat{\pi}(\mathbf{X}_i)} \right) (Y_i - \alpha - \tau D_i)^2$$

Variance

- If $\widehat{\pi}(\mathbf{X}_i)$ is estimated, how to estimate $\mathbb{V}[\widehat{\tau}_{\text{ipw}}]$ or $\mathbb{V}[\widehat{\tau}_h]$?
- First option: **bootstrap** (possibly with clustering if needed)
- Second option: **method of moments** (Newey and McFadden, 1994)
  - Treat this a joint estimation problem and use the delta method.
  - Moment conditions for the propensity score model with parameters θ:

$$\mathbb{E}\left[\underbrace{\left(\frac{D_i}{\pi_\theta(\mathbf{X}_i)} - \frac{1-D_i}{1-\pi_\theta(\mathbf{X}_i)}\right)\frac{\partial \pi_\theta(\mathbf{X}_i)}{\partial \theta}}_{\text{score for treatment model}}\right] = 0$$

  - Moment conditions for weighting estimators:

$$\text{HT: } \mathbb{E}\left[\frac{D_i Y_i}{\pi_\theta(\mathbf{X}_i)} - \mathbb{E}[Y_i(1)]\right] = \mathbb{E}\left[\frac{(1-D_i)Y_i}{1-\pi_\theta(\mathbf{X}_i)} - \mathbb{E}[Y_i(0)]\right] = 0$$

$$\text{Hajek: } \mathbb{E}\left[\frac{D_i(Y_i - \mathbb{E}[Y_i(1)])}{\pi_\theta(\mathbf{X}_i)}\right] = \mathbb{E}\left[\frac{(1-D_i)(Y_i - \mathbb{E}[Y_i(0)])}{1-\pi_\theta(\mathbf{X}_i)}\right] = 0$$

- Replace with sample versions and use delta method to get asymptotic variance.

Estimated versus known pscores

```
ht.est <- function(y, d, w) {
  n <- length(y)
  (1/n) * sum((y * d * w) - (y * (1-d) * w))
}
n <- 200
x <- rbinom(n, size = 1, prob = 0.5)
dprobs <- 0.5*x + 0.4*(1-x)
d <- rbinom(n, size = 1, prob = dprobs)
y <- 5 * d - 10 * x + rnorm(n, sd = 5)

true.w <- ifelse(d == 1, 1/dprobs, 1/(1-dprobs))
pprobs <- predict(glm(d ~ x))
est.w <- ifelse(d == 1, 1/pprobs, 1/(1 - pprobs))
ht.est(y, d, est.w)
```

```
## [1] 5.22
```

```
ht.est(y, d, true.w)
```

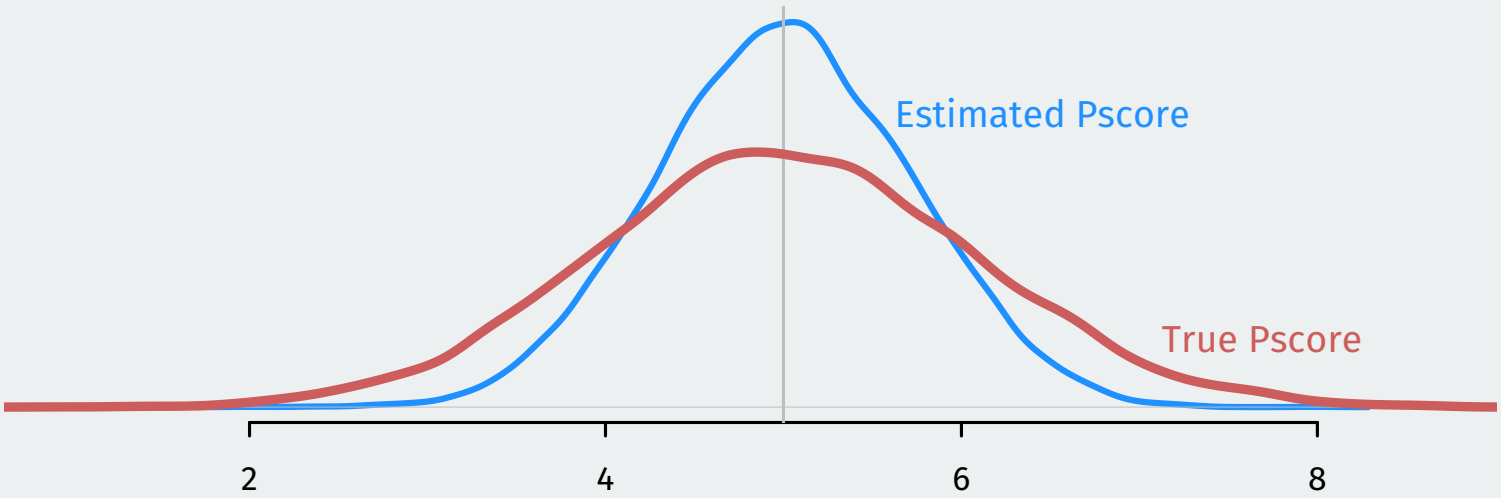
```
## [1] 5.56
```

Sampling distribution of the HT estimators

```
sims <- 10000
true.holder <- rep(NA, sims)
est.holder <- rep(NA, sims)
for (i in 1:sims) {
  x <- rbinom(n, size = 1, prob = 0.5)
  dprobs <- 0.5*x + 0.4*(1-x)
  d <- rbinom(n, size = 1, prob = dprobs)

  y <- 5 * d - 10 * x + rnorm(n, sd = 5)
  true.w <- ifelse(d == 1, 1/dprobs, 1/(1-dprobs))
  pprobs <- predict(glm(d ~ x))
  est.w <- ifelse(d == 1, 1/pprobs, 1/(1 - pprobs))
  est.holder[i] <- ht.est(y, d, est.w)
  true.holder[i] <- ht.est(y, d, true.w)
}
```

Sampling distribution of the HT estimators




```
var(est.holder)
```

```
## [1] 0.506
```

```
var(true.holder)
```

```
## [1] 1.15
```

Why use estimated pscores?

- Why is the estimated propensity score more efficient than the true PS?
- **Removing chance variations** using $\widehat{\pi}(\mathbf{X}_i)$ adjusts for any small imbalances that arise because of a finite sample.
- True PS only adjusts for the **expected** differences between samples.
- Only true if propensity score model is **correctly specified!!**

Augmented IPW estimator

- **Augmented** IPW estimator combines regression and weighting:

$$\begin{aligned}\widehat{\tau}_{\text{aipw}} &= \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{D_i Y_i}{\widehat{\pi}(\mathbf{X}_i)} - \frac{(1-D_i)Y_i}{1-\widehat{\pi}(\mathbf{X}_i)}\right. \\ &\qquad\left. - \left(\frac{D_i - \widehat{\pi}(\mathbf{X}_i)}{\widehat{\pi}(\mathbf{X}_i)}\hat{\mu}_1(\mathbf{X}_i) - \frac{D_i - \widehat{\pi}(\mathbf{X}_i)}{1-\widehat{\pi}(\mathbf{X}_i)}\hat{\mu}_0(\mathbf{X}_i)\right)\right\} \\ &= \frac{1}{n}\sum_{i=1}^{n}\left\{\hat{\mu}_1(\mathbf{X}_i) - \hat{\mu}_0(\mathbf{X}_i)\right. \\ &\qquad\left. + \left(\frac{D_i(Y_i - \hat{\mu}_1(\mathbf{X}_i))}{\widehat{\pi}(\mathbf{X}_i)} - \frac{(1-D_i)(Y_i - \hat{\mu}_0(\mathbf{X}_i))}{1-\widehat{\pi}(\mathbf{X}_i)}\right)\right\}\end{aligned}$$

- **Double robustness**: consistent if either $\widehat{\pi}(\mathbf{X}_i)$ or $\hat{\mu}_d(\mathbf{X}_i)$ are consistent.
  - Can allow each model to be more flexible without hurting asymptotics.
- **Efficient**: lowest asymptotic variance among consistent estimators when PS model is correct.

Problems with weighting

- Propensity score balances covariates across **repeated samples**
    - Relies on the law of large numbers (not optimized for small samples)
- Difficult to model propensity score for covariate balance.
- Highly variable/unstable weights $\rightsquigarrow$ high variance estimators.
    - When there is a lack of overlap so $\pi(\mathbf{X}_i)$ is close to 0 or 1.
    - **Windsorizing**: trim weights beyond 5th and 9th percentile

Covariate balancing propensity scores

- How to help with these issue: prioritize balance in estimation of $\pi(\mathbf{X}_i)$
  - One approach: **covariate balancing propensity scores** (Imai and Ratkovic)
- Usual maximum likelihood estimation solves a moment condition:

$$\mathbb{E}\left[\underbrace{\left(\frac{D_i}{\pi_\theta(\mathbf{X}_i)} - \frac{1-D_i}{1-\pi_\theta(\mathbf{X}_i)}\right)\frac{\partial \pi_\theta(\mathbf{X}_i)}{\partial \theta}}_{\text{score}}\right] = 0$$

- CBPS adds balancing conditions:

$$\mathbb{E}\left[\left(\frac{D_i}{\pi_\theta(\mathbf{X}_i)} - \frac{1-D_i}{1-\pi_\theta(\mathbf{X}_i)}\right) f(\mathbf{X}_i)\right] = 0$$

  - If $f(\mathbf{X}_i) = \mathbf{X}_i$, then condition says the means must be balanced.
  - If $f(\mathbf{X}_i) = \mathbf{X}_i^2$, then balances the second moments.
- Estimation by generalized method of moments (GMM).
  - More moment conditions than parameters $\rightsquigarrow$ can't satisfy all conditions.
  - Find parameters that get closest to satisfying sample moment

Balancing weights

- Alternative: find weights that directly balance covariates.
- **Stable balancing weights** (Zubizarreta, 2015) one recent example.
    - Rearrange data so first m units are controls, rest are treated.
- Solve the following convex quadratic programming problem:

$$\min_{w_1,\ldots,w_m} \sum_{i=1}^{m} (w_i - \overline{w})^2$$

$$\text{such that} \quad \sum_{i=1}^{m} w_i = 1, \quad w_i \geq 0, \quad \left| \frac{1}{n_1} \sum_{i:D_i=1} X_{ik} - \sum_{i=1}^{m} w_i X_{ik} \right| \leq \delta_k$$

- Minimum variance weights that approximately balance covariates.
    - Amount of allowed imbalance, δ_k, selected by researcher.
    - Can include X_{ik}^2 etc to balance other parts of the distribution.